

New More Docker. Easy Access. New Streamlined Plans. Learn more. →

docker EARLY ACCESS

Settings / Personal access tokens / New access token

Copy access token

Use this token as a password when you sign in from the Docker CLI client. [Learn more](#)

Make sure you copy your personal access token now. Your personal access token is only displayed once. It isn't stored and can't be retrieved later.

Access token description
new

Access permissions
Read, Write, Delete

To use the access token from your Docker CLI client:

1. Run

```
$ docker login -u amitow
```

[Copy](#)
2. At the password prompt, enter the personal access token.

amitow

- What's new
- My profile
- Account settings**
- Billing
- Sign out

New Access Token

A personal access token is similar to a password except you can have many tokens and revoke access to each one at any time. [Learn more](#)

Access Token Description *

jenkins

Access permissions

Read, Write, Delete

Read, Write, Delete tokens allow you to manage your repositories.

Cancel

Generate

- In your Jenkins server go to Manage Jenkins -> Credentials -> System -> Global credentials -> + Add credentials. Enter your Docker Hub username, the token you created and a brief description of the credentials.

New credentials

Kind

Username with password ▼

Username ?

83910336618

☒ Treat username as secret ?

Historically, not only passwords but usernames were masked in the build log. Since these can interfere with diagnostics, and cause unrelated occurrences of a common word to be masked, you may choose to leave usernames unmasked if they are not sensitive. Note that regardless of this setting, the username will be displayed to anyone permitted to reconfigure the credentials. (from [Credentials Plugin](#))

Password ?

.....

ID ?

Description ?

Dockerhub credentials

Create

Create the environment variables used in the Jenkinsfile for the Docker image and for the Docker Hub credentials.

- Go to Manage Jenkins -> Configure System and under 'Global properties' add them as shown in the picture below.
- Name: DOCKER_BFLASK_IMAGE
- Value: [your-dockerhub-username/repository-name]
- Name: DOCKER_REGISTRY_CREDS
- Value: ID generated when adding the Docker Hub credentials into Jenkins.

Dashboard > Manage Jenkins > Configure System >

Global properties

☐ Disable deferred wipeout on this node ⓘ

☒ Environment variables ⓘ

List of variables ⓘ

Name	Value
DOCKER_BFLASK_IMAGE	8991033661b/basic_flask_app
DOCKER_REGISTRY_CREDS	480w0646-9310-4d67-8a8e-6d29a1111251

Docker environment credentials

Create a new Jenkins job.


- In your Dashboard go to New Item -> Pipeline and enter the name of your project.


Dashboard >


Enter an item name


My flask app


» Required field



Freestyle project
 This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.


Pipeline
 Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.


Multi-configuration project
 Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.


Folder
 Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.


Multibranch Pipeline
 Creates a set of Pipeline projects according to detected branches in one SCM repository.



Organization Folder
 Creates a set of multibranch project subfolders by scanning for repositories.


Jenkins' projects


- Select 'Github project' and add your Github repository.

Dashboard > pipeline-job > Configuration

Configure

 General


 Advanced Project Options


 Pipeline


☐ Do not allow concurrent builds


☐ Do not allow the pipeline to resume if the controller restarts

☒ GitHub project

Project url 

Advanced 

☐ Pipeline speed/durability override 

☐ Preserve stashes from completed builds 

Jenkins pipeline configuration

- In the Pipeline section, under SCM select 'Git' and enter your repository again.
- **Note:** In case it is a private repository you would need to provide your Github credentials.

Git 



Repositories 

Repository URL 



Credentials 


- none - 


+ Add 

Advanced 

Add Repository

Jenkins SCM configuration

Script Path 

☒ Lightweight checkout 

[Pipeline Syntax](#)

Save

Apply